Understanding speech based on a Bayesian concept extraction method

Salma Jamoussi, Kamel Smaïli, Jean-Paul Haton

Understanding speech based on a Bayesian concept extraction method

Salma Jamoussi, Kamel Smaïli, and Jean-Paul Haton

LORIA/INRIA-Lorraine
615 rue du Jardin Botanique, BP 101, F-54600 Villers-lès-Nancy, France
{jamoussi, smaili, jph}@loria.fr

Abstract. The automatic speech understanding problem could be considered as an association problem between two different languages. At the entry, the query expressed in oral or written natural language and at the end, just before the interpretation stage, the same request is expressed in term of concepts. One concept represents a given meaning, it is defined by a set of words sharing the same semantic properties. In this paper, we propose a new Bayesian network based method to automatically extract the underlined concepts. We also propose three different approaches for the vector representation of words. This representation allows the Bayesian network to build the adequate list of concepts for the considered application. This step is very important to obtain well built concepts. We finish this paper by a description of the post-processing step during which, we label our sentences and we generate the corresponding SQL queries. This step allows us to validate our automatic understanding approach and to obtain 92.5% of correct SQL queries on the test corpus.

1 Introduction

Interactive applications must be able to process users spoken queries. It means they have to recognize what has been uttered, extract its meaning and give suitable answers or execute right corresponding commands. In such applications, the speech understanding component constitutes a key step. Several methods were proposed in the literature to clean up this problem and the majority of them is based on stochastic approaches for conceptual decoding. These methods allow to reduce the need of human expertise, however they require a supervised learning step which means a former stage of manual annotation of the training corpus [1, 4, 5].

The data annotation step consists in segmenting the data into conceptual segments where each segment represents an underlined meaning [1]. Within this step, we have to find first of all the list of concepts which are related to the considered corpus. Then, we can use these concepts to label the segments of each sentence in the corpus and finally, we can launch the training step. Doing all this in a manual way constitutes a tiresome and an expensive phase. Moreover, the manual extraction is prone to subjectivity and to human errors. Automating

this task will thus reduce the human intervention and will especially allow us to use the same process when context changes. Our purpose in this paper is to fully automate the understanding process from the input signal until the SQL request generation step.

In the following, we start by describing the general architecture of our understanding system based on the approach suggested in [5]. Then, we present a new approach to automatically extract the semantic concepts of the considered application using a Bayesian network for unsupervised classification, called AutoClass. For this, three different methods for the vector representation of words are exposed, these representations will help the Bayesian network to build concepts. Finally, we will describe the last stage of our understanding process, in which we label the user requests and we generate the associated SQL queries.

2 Automatic speech understanding

A speech understanding system could be considered as a machine that produces an action as the result of an input sentence. Thus, the understanding problem could be seen as a translation process, it translates a signal (represented by a sequence of words) into a special form that represents the meaning convoyed by the sentence. In a first time, the sentence is labelled by a list of conceptual entities (often called concepts), these labels constitute a useful intermediate representation which must be simple and representative. In a second time, this representation will be used to interpret semantically the sentence.

The speech understanding problem can be seen then as an association problem, where we have to associate inputs (e.g. speech or text) to their respective meanings represented by a list of concepts. A concept is related to a given meaning, it is given by a set of words expressing the same idea and sharing the same semantic properties. For example, the words *plane, train, boat, bus* can all correspond to the concept "*transport means*" in a travel application.

The step of interpretation consists in converting the obtained concepts to an action to be done as a final response to the user. In order to achieve such a goal, we have to convert these concepts into a target formal command (e.g. an SQL query, a shell command, etc.). The figure 1 illustrates the general architecture of such speech understanding system, this model was given in [5] and it was included in several other works because of its effectiveness and its simplicity [4, 1]. We also, adopt the same general architecture but we propose new techniques within each component.

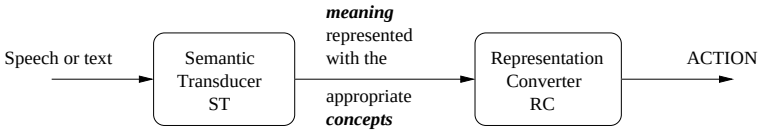

Fig. 1. General architecture of a speech understanding system.

In our work, we are interested by a bookmark consultation application, for that we use the corpus of the European project MIAMM. The aim of this project is to build up a platform of an oral multimodal dialogue. The corpus contains 71287 different queries expressed in French. Each query expresses a particular manner to request the database. Some examples of these queries are given in the table 1. These queries are provided to the understanding system in their textual form. Our goal is to provide at the end the corresponding SQL query which can answer the user request.

Table 1. Some examples of queries in the MIAMM corpus.

Show me the contents of my bookmarks.
I would like to know if you can take the contents that I prefer.
Do you want to select the titles that I prefer.
Is it possible that you select the first of my bookmarks.
Is it possible to indicate me a similar thing.
Can you show me only December 2001.
It is necessary that you print the list that I used early this morning.
I want to see the second that I looked at in the morning.

3 Concepts extraction : methods and results

The aim of this step is to identify the semantic concepts related to our application. The manual determination of these concepts is a very heavy task, so we should find an automatic method to achieve such a work. The automatic method could give worse results than those obtained by the manual way, but it allows a complete automation of the understanding process. The method to be used must be able to gather the words of the corpus in various classes in order to build up the list of appropriate concepts.

To reach our goal we used an unsupervised classification technique. Among the unsupervised classification methods, we tried the Kohonen maps, the Oja and Sanger neural networks, the K-means method and some other methods based on the mutual information measure between words [3]. The obtained concepts were quite significant, but contained a lot of "noise", it means that we found many words which did not have their place in the meaning expressed by these concepts. To solve this problem, we explored other methods and adopted the Bayesian network technique because of its mathematical base and its powerful inference mechanism. We use then, AutoClass a Bayesian network for unsupervised classification, it accepts real and discrete values as input. As result, it provides for each input, its membership probabilities in all the found classes. AutoClass is based on the Bayes theorem and it supposes that there is a hidden multinomial variable which represents the various classes of the input data. More mathematical details about this software can be found in [2].

In the next sections we present three different approaches to represent words in vectorial aspect. This representation, which must be semantically significant, constitutes a key stage in the understanding process. In fact, according to this representation, the Bayesian network will decide of words to group in the same class in order to build up the needed list of concepts.

3.1 The word context

One word can have several features but only few of them are relevant for a good semantic representation. In a first step, we decided to associate to each word its different contexts. We consider that if two words have the same contexts then they are semantically similar. In this approach, a word will be represented by a vector of $2 \times N$ elements containing the N left context words and the N right context words. Figure 2 shows how we associate for each word its left and right bigram contextual representation.

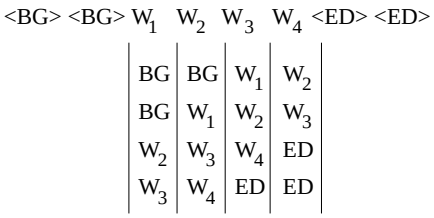

Fig. 2. The bigram contextual representation of words.

The obtained classes represent many good semantic concepts, but an important overlapg has been noticed. Moreover, we had difficulties in controlling the number of concepts. Some examples of these concepts are given in table 2.

Table 2. Some obtained concepts by using the bigram contextual representation.

Concept	Group of words
Bookmarks_1	Preferred, favourites, chosen, appreciated, liked, adored
Bookmarks_2	Favourites, preferred, listened, seen, used, looked at
Bookmarks_3	Favourites, preferred, chosen, appreciated, liked, adored, similar, same, equivalent, resembling, synonymous, near, identical, close
Request_1	Possible, request, wants, would like, like, wishes, would wish
Request_2	Can, could, wants, like, possible, request, would like, want, would wish, is necessary, wishes
Order	Show, indicate, select, find, give, post, press, take, pass, seek

3.2 Similarity between words

To find more homogeneous concepts, we completely changed the vector structure of each word. We used the average mutual information measure which tries to find contextual similarities between words.

In this approach, we associate to each word a vector with M elements, where M is the size of the lexicon. The jth element of this vector represents the average mutual information between the word number j of the lexicon and the word to be represented (equation 1).

$$W_i = \left[I(w_1 : w_i), I(w_2 : w_i), \ldots, I(w_j : w_i), \ldots, I(w_M : w_i)\right] \tag{1}$$

This vector expresses the similarity degree between the word to represent and all the other words of the corpus. The formula of the average mutual information between two words w_a and w_b is given by :

$$\begin{aligned} I(w_a : w_b) = P(w_a, w_b) \log \frac{P(w_a|w_b)}{P(w_a)P(w_b)} + P(w_a, \overline{w}_b) \log \frac{P(w_a|\overline{w}_b)}{P(w_a)P(\overline{w}_b)} + \\ P(\overline{w}_a, w_b) \log \frac{P(\overline{w}_a|w_b)}{P(\overline{w}_a)P(w_b)} + P(\overline{w}_a, \overline{w}_b) \log \frac{P(\overline{w}_a|\overline{w}_b)}{P(\overline{w}_a)P(\overline{w}_b)} \end{aligned} \tag{2}$$

Where $P(w_a, w_b)$ is the probability to find w_a and w_b in the same sentence, $P(w_a \mid w_b)$ is the probability to find w_a knowing that we already met w_b, $P(w_a)$ is the probability of the word w_a and $P(\overline{w}_a)$ is the probability of any other word except w_a.

By using this vector representation, the Bayesian network achieves homogeneous semantic classes. A class is made up of words sharing the same semantic properties. The number of classes is very coherent with our application. This representation also enables us to solve the problem of the overlapping between concepts. In the table 3, we give some examples of the obtained concepts, where one can notice that there is no overlapping. However we still have some imperfections as in the case of the concepts *Request_1, Request_2* and *Request_3* which should be gathered in the same class.

Table 3. Some examples of concepts obtained by using the representation based on the average mutual information measure.

Concept	Group of words
Bookmarks	Favourites, preferred, chosen, appreciated, adored
Way	Listened, seen, looked at, used
Similarity	Similar, equivalent, resembling, synonymous, near, identical, close
Request_1	Could, want, would like
Request_2	Possible, would like, would wish
Request_3	Wish, is necessary, wishes, would wish
Order	Show, indicate, select, find, give, post, present, take, pass, seek

3.3 Combinaison : context and similarity

In this approach we combined the two preceding representations in order to improve results. In the first approach we work on the occurrence level where we directly exploit information related to the word context. In the second one, we use a measure to seek for similarities between words. We can easily notice that the information used in these two methods is different but complementary.

To combine these two methods, we decided to represent each word by a matrix $M \times 3$ of average mutual information measures. The first column of this matrix corresponds to the preceding vector of average mutual information (see section 3.2), the second column represents the average mutual information measures between the vocabulary words and the left context of the word to be represented. The third column is determined by the same manner but it concerns the right context. The jth value of the second column is the weighted average mutual information between the jth word of the vocabulary and the vector constituting the left context of the word W_i. It is calculated as follows :

$$IMM_j(C_l^i) = \frac{\sum_{w_l \in \, left\ context\ of\ W_i} I(w_j : w_l) \times K_{wl}}{Nb\_occ} \tag{3}$$

Where $IMM_j(C_l^i)$ is the average mutual information between the word w_j of the lexicon and the left context of the word W_i. $I(w_j : w_l)$ represents the average mutual information between the word number j of the lexicon and the word w_l which belongs to the left context of the word W_i. K_{wl} is the number of times where the word w_l is found in the left context of the word W_i and Nb_occ is the total number of occurrences of the word W_i in the corpus. The word W_i thus represented by the matrix shown in the figure 3.

$$W_i = \begin{bmatrix} I(w_1 : w_i) & IMM_1(C_l^i) & IMM_1(C_r^i) \\ I(w_2 : w_i) & IMM_2(C_l^i) & IMM_2(C_r^i) \\ \vdots & \vdots & \vdots \\ I(w_j : w_i) & IMM_j(C_l^i) & IMM_j(C_r^i) \\ \vdots & \vdots & \vdots \\ I(w_M : w_i) & IMM_M(C_l^i) & IMM_M(C_r^i) \end{bmatrix}$$

Fig. 3. Representation of the word W_i by the combined method.

The matrix used to represent a word in the corpus exploits a maximum number of information that can be related to this word. It considers its context and its similarity with all the other words of the lexicon. Such a word representation could help the Bayesian network to classify the words and allows us to considerably improve results. We obtain a coherent list of concepts. We decided to keep these ones for the rest of the understanding treatment. Some examples of these results are given in the table 4.

Table 4. Some examples of concepts obtained by using the combined representation.

Concept	Group of words
Bookmarks	Favourites, preferred, chosen, appreciated, adored, liked
Way	Listened, seen, looked at, used
Similarity	Similar, equivalent, resembling, synonymous, near, identical, close
Request	Wish, wishes, would wish, can, wants, like, possible, would like
Order	Show, indicate, select, find, give, post, present, take, pass, seek

4 Labelling and postprocessing

The last step consists in providing the SQL queries associated with the input textual requests. During this phase, we start by the request interpretation. In fact, if we have all the concepts which govern our application, we can affect to each query the suitable concepts. This is the semantic translation stage, the first component of the general architecture of our understanding system (see the figure 1). Within this step, we only need to label our data by associating to each word in the sentence its corresponding semantic class. Since our concepts do not overlap, labelling the requests does not present any risk of ambiguity.

Then, we can pass to the second component of our model, the "Representation Converter", where we have to convert the found concepts into SQL queries which allow us to extract the necessary information from our data base. For this reason we implemented an inference engine which associates to each concept one or more generic sub-queries. In a generic SQL query, the concepts take the places of the conditions. For example, if we find the concept "Date", we don't know the value of this date but, we can indicate in the generated query that there is a condition on the date. This inference engine takes into account the repetitions, the lapses, the multiple and the implicit requests and the others phenomena of the spontaneous speech. In the following phase, we instantiate each concept, in the generic request, by its value which is deduced by going back to the initial sentence. At the end we obtain a well formed SQL query that we can carry out to extract the required bookmarks. Obtained results are very encouraging, in fact, in term of correct SQL queries, we obtain a rate of 100% with the training corpus and a rate of 92.5% with a test corpus containing 400 sentences. In figure 4, we give an example illustrating the various stages followed in order to generate a good SQL query.

5 Conclusion

In this article, we consider that the automatic speech understanding problem can be seen as an association problem between two different languages, the natural language and the concept language. Concepts are semantic entities gathering a set of words which share the same semantic properties and which express a given idea. We proposed three different methods to automatically extract the concepts

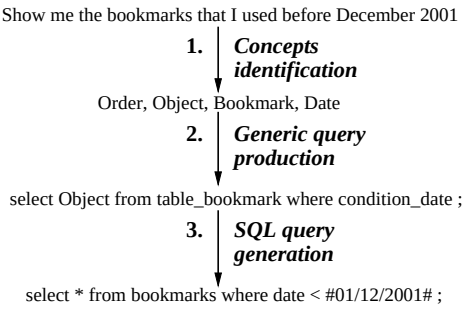

Fig. 4. Treatment sequence : from a natural language request to the corresponding SQL query.

using a Bayesian network, as well as an approach for automatic sentence labelling and an engine for generating SQL queries corresponding to the user requests.

The concept extraction and the sentence labelling tasks are usually carried out manually. They constitute then, the most delicate and the most expensive phase in the understanding process. The methods suggested in this article allow us to avoid the need for the human expertise and can be used for several other research fields which use the semantic classification : text categorization, information retrieval and data mining. The most crucial problem was how to represent adequately the words in view of a good clustering. The combined method gave the best results thanks to the information that it uses to represent a word in the corpus. For our application of bookmark pages consultation, the obtained concepts were very efficient. They allowed us then, to achieve the stage of labelling without any major difficulties and to obtain good results in terms of concepts viability and relevant retrieved SQL requests. In fact, with the test corpus, we obtain a rate of 92.5% of correct SQL queries. As future work we plan to extend the postprocessing module to make it able to react vis-a-vis new key words not included in the concepts and to integrate our understanding module is SIROCCO, a French speech regonition system [6].

References

1. Bousquet-Vernhettes C., Vigouroux N.: Context use to improve the speech understanding processing. Int. Workshop on Speech and Computer, Russia (2001)
2. Cheeseman P., Stutz J.: Bayesian classification (AutoClass): theory and results. Advances in Knowledge Discovery and Data Mining (1996)
3. Jamoussi S., Smaïli K., Haton J.P.: Neural network and information theory in speech understanding. Int. Workshop on Speech and Computer, Russia (2002)
4. Lefèvre F., Bonneau-Maynard H.: Issues in the development of a stochastic speech understanding system. Int. Conf. on Spoken Language Processing, Denver (2002)
5. Pieraccini R., Levin E., Vidal E.: Learning how to understand language. Proc. 4rd European Conf. on Speech Communication and Technology, Germany (1993)
6. Gravier G., Yvon F., Jacob B., Bimbot F.: Sirocco - un systme ouvert de reconnaissance de la parole. Journes d'Etudes sur la Parole, France (2002).